

한이음 공모전 2017 참가신청서

작품 정보

프로젝트명	국문	[17-개발-P179] SW품질향상을 위한 아키텍처표준 인스펙션 개발 프로젝트
	영문	StyleIDE with Code Convention and Code Review
프로젝트기간	2017. 3. 27. ~ 2017. 11. 30.	
작 품 명	StyleIDE	
작품소개	코드 컨벤션 및 코드 리뷰 기능이 내장된 IDE	
주제영역	<input type="checkbox"/> 건강 <input type="checkbox"/> 생산성 <input type="checkbox"/> 생활 <input type="checkbox"/> 안전 <input type="checkbox"/> 엔터테인먼트 <input checked="" type="checkbox"/> 기타 ()	
타 대회참가 신청수상여부	<input checked="" type="checkbox"/> 미참가 <input type="checkbox"/> 참가신청 중 <input type="checkbox"/> 수상	

팀 정보

팀 명		[17-개발-P179] SW품질향상을 위한 아키텍처표준 인스펙션 개발 프로젝트			
팀 원	이 름	소 속	부서/학과	직위/학년	
멘 토	최금열	우리에프아이에스	품질정책팀	팀장	
지도교수					
멘 티 (참여학생)	멘티 1(팀장)	문성욱	부산대학교	정보컴퓨터공학부	4
	멘티 2	박문수	부산대학교	정보컴퓨터공학부	4
	멘티 3	권성욱	부산대학교	정보컴퓨터공학부	4
	멘티 4	박슬기	홍익대학교	컴퓨터공학부	4
	멘티 5	최효진	홍익대학교	컴퓨터공학부	4

본인은 「한이음 공모전 2017」의 제반규정 및 유의사항을 준수하고 제출된 서류의 모든 내용에 허위 사실이 없음을 서약합니다. 또한, 공모전 심사를 위한 평가에 성실히 응할 것이며 참가자와 관련된 정보 활용에 동의합니다.

★ 개인정보 수집·이용(개인정보보호법 제15조)

- * [수집·이용목적] 한이음 공모전 및 한이음 엑스포 행사 운영/관리, 특허출원을 위한 선행기술조사, 언론홍보 및 행사안내, 한이음 사이트 등 사업 관련 자료 공개 및 홍보자료 활용
- * [수집항목] 이름, 소속, 부서/학과, 직위/학년, 전화번호, 이메일, 프로젝트 수행내용
- * [보유·이용기간] 사업 종료 후 5년

선정된 작품은 「한이음 엑스포 2017」 행사에 반드시 전시해야 하며, 전시 불참 시 평가 및 수상에서 제외됨을 확인하고 본 공모전에 참가 신청합니다.

본 참가신청서 제출 시, 위 모든 사항에 동의한 것으로 간주합니다.

2017년 9월 3일

[붙임] 개발보고서 1부

한이음 공모전 2017 개 발 보 고 서

2017. 9. 3

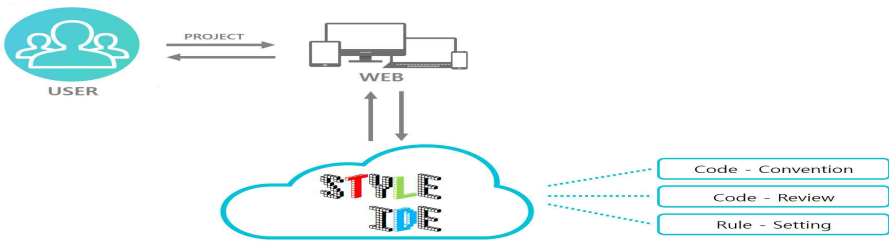
프로젝트명	국문	[17-개발-P179] SW품질향상을 위한 아키텍처표준 인스펙션 개발 프로젝트
	영문	StyleIDE with Code Convention and Code Review
작 품 명	StyleIDE	
신 청 자	부산대학교 문성욱	

요 약 본

팀 정보

팀 명	[17-개발-P179] SW품질향상을 위한 아키텍처표준 인스펙션 개발 프로젝트			
팀 원	이 름	소 속	부서/학과	직위/학년
멘 토	최금열	우리에프아이에스	품질정책팀	팀장
지도교수				
멘티 1(팀장)	문성욱	부산대학교	정보컴퓨터공학부	4
멘티 2	박문수	부산대학교	정보컴퓨터공학부	4
멘티 3	권성욱	부산대학교	정보컴퓨터공학부	4
멘티 4	박슬기	홍익대학교	컴퓨터공학부	3
멘티 5	전호진	홍익대학교	컴퓨터공학부	3



작품 정보		
프로젝트명	국문	[17-개발-P179] SW품질향상을 위한 아키텍처표준 인스펙션 개발 프로젝트
	영문	StyleIDE with Code Convention and Code Review
작품명	StyleIDE	
작품 소개	<p>사용자의 코드 작성 스타일을 가이드 해 줄수 있는 코드 컨벤션 및 코드 리뷰를 할 수 있는 도구들이 내장된 IDE.</p> <p>클라우드 상에서 실행되어 설치없이 사용자가 편리하게 이용가능하다.</p>	
작품 구성도	 <pre> graph LR USER((USER)) <--> PROJECT WEB[WEB] WEB <--> CLOUD((STYLE IDE)) CLOUD -.-> CC[Code - Convention] CLOUD -.-> CR[Code - Review] CLOUD -.-> RS[Rule - Setting] </pre>	
작품의 개발배경 및 필요성	<ul style="list-style-type: none"> ○ 소프트웨어 개발에 있어서 SW품질관리는 매우 중요. ○ 코딩을 공부하는 학생들의 코딩 스타일을 준수하도록 유도. 	
작품의 특징점	<p>기존의 코드 관리프로그램 sonarqube는 프로젝트와 연동하려면 복잡한 설정과정이 필요, 하지만 이러한 기능들을 내장함으로서 사용자가 쉽게 이용하도록 편의성 제공.</p>	
작품 기능	<p>프로젝트 관리 : 웹 페이지 기반 프로젝트 관리</p> <p>코드 컨벤션 수행 : 표준 코드 스타일을 지키도록 유도</p> <p>코드 리뷰 수행 : 자신의 코드를 다른 사람들에게 평가 받을수 있도록 함</p> <p>코드 공유 : 하나의 코드를 여러 사람들과 공유 함으로서 실시간 협업 유도</p>	
작품의 기대효과 및 활용분야	<ul style="list-style-type: none"> ○ 이용자가 표준 코드 스타일 가이드를 지키도록 유도. ○ 사용자의 코드 스타일을 분석 후 통계 자료를 지속적으로 보고. 	

본 문

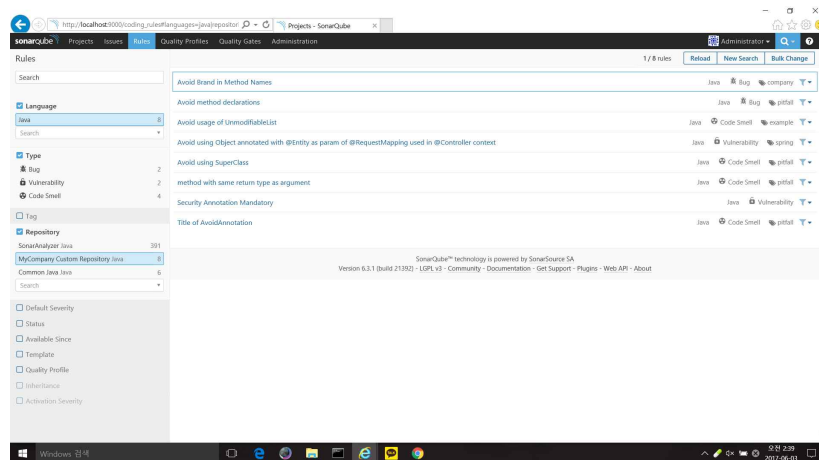
I. 작품 개요

※ 평가항목 : 기획력 (필요성, 차별성)

1. 작품 소개

○ 기획의도

- 소프트웨어 개발에 있어서 인스펙션은 매우 중요함
- 코딩을 배우는 학생들이 쉽고 간단하게 이용할 수 있는 품질관리 프로그램 개발을 목표로 함



[그림] 품질관리도구중 하나인 ‘소나큐브’를 설정 화면

○ 작품내용

- 사용자가 개발한 프로젝트에 적용할 룰을 선택하면 그에 따른 인스펙션 진단결과를 보여줌.
- 룰 설정, 리뷰 등의 기능에 권한을 설정하여 팀 프로젝트를 관리 할 수 있음.
- 프로젝트 멤버, 이슈, 일정 관리 등을 할 수 있음.

2. 작품의 개발 배경 및 필요성

- 소프트웨어 개발에 있어서 SW품질관리는 매우 중요.
- 코딩을 공부하는 학생들의 코딩 스타일을 준수하도록 유도.

3. 작품의 특징 및 장점

- 쉽고 다양한 룰 선택

기존의 eclipse, visual studio 등의 프로그램도 코드 컨벤션을 제공함. 하지만 사용자가 원하는 룰만을 적용시키려면 일련의 과정이 필요함. 본 작품은 사용자가 쉽게 다양한 룰을 선택하고, 그를 적용시켜 코드 컨벤션을 실행 할 수 있음. 사용자 맞춤 컨벤션이 가능하여 편의를 제공함.

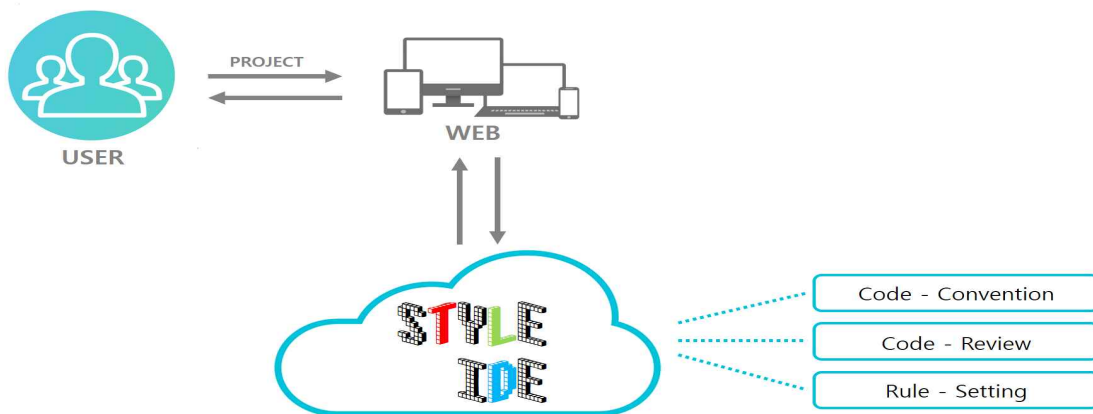
- 간단한 리뷰절차

: 기존의 코드 리뷰프로그램 Gerrit은 git저장소와 연동하여 사용하기 때문에 git에 대한 이해가 필요함. 반면에 본 작품은 절차가 간단해 초보개발자들의 팀 프로젝트, 코딩교육을 할 때 유용하게 사용할 수 있음.

II. 작품 내용

※ 평가항목 : 기술력 (기능구체성, 난이도, 완성도)

1. 작품 구성도



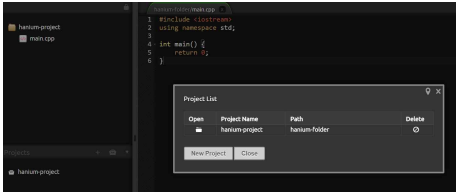
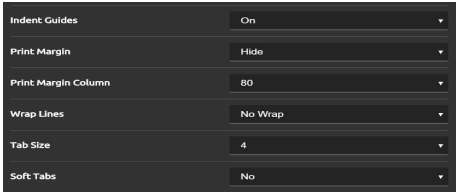
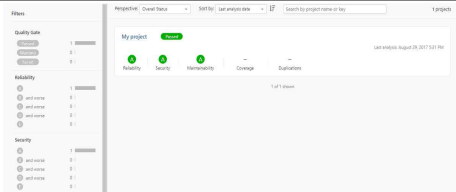

- 코드 컨벤션 및 리뷰 기능을 내장한 웹 IDE를 통해 프로젝트를 작성 및 관리 한다.

2. 작품 기능

2-1. 전체 기능 목록

구분	기능	설명	현재진척도(%)
S/W	회원가입	프로그램 사용을 원하는 사용자에게 회원 가입을 제공한다.	100 %
	로그인 / 로그아웃	등록된 사용자에게 로그인/ 로그아웃을 처리한다.	100 %
	사용자 등록정보 관리	로그인한 사용자에게 개인 정보 관리 기능을 제공한다.	100 %
	프로젝트 관리	사용자에게 새로운 프로젝트에 대하여 CRUD 및 공유기능을 수행한다. (9월 말 완성 예정)	50 %
	이슈 관리	프로젝트에 대하여 이슈(프로젝트 상의 문제) 발생 시 CRUD 및 공유기능을 수행한다. (9월 말 완성 예정)	60 %
	멤버 관리	프로젝트와 관련된 멤버에 대하여 관리 (9월 말 완성 예정)	80 %
	일정 관리	프로젝트 진행에 대한 일정 관리를 제공한다. (9월 말 완성 예정)	60 %
	룰 관리	코드 컨벤션에서 사용할 룰 라이브러리에 대해 확인 및 설정이 가능하다. (10월 중순 완성 예정)	50 %
	코드 컨벤션	주어진 프로젝트 및 소스에 대하여 지정된 룰을 이용하여 코드를 검증한다. (10월 중순 완성 예정)	50 %
	코드 리뷰	주어진 소스를 확인하고 코멘트를 남긴다. (9월 말 완성 예정)	70 %

2-2. S/W 주요 기능

기능	설명	작품실물사진
프로젝트 관리	사용자에게 새로운 프로젝트에 대하여 CRUD 및 공유기능을 수행한다.	
이슈 관리	프로젝트에 대하여 이슈(프로젝트 상의 문제) 발생 시 CRUD 및 공유기능을 수행한다.	미완성
룰 관리	코드 컨벤션에서 사용할 룰 라이브러리에 대해 확인 및 설정이 가능하다.	
코드 컨벤션	주어진 프로젝트 및 소스에 대하여 지정된 룰을 이용하여 코드를 검증한다.	
코드 리뷰	주어진 소스를 확인하고 코멘트를 남긴다.	

- Syntax Highlight 기능은 Opensource Project인 ACE 사용.
- IDE 기능은 Opensource Project인 Codiad를 기반으로 제작.

2-3. H/W 주요 기능

기능/부품	설명	작품실물사진

3. 주요 적용 기술

S/W 기능 구현을 위한 시나리오

-프로젝트 관리

사용자에게 새로운 프로젝트에 대하여 CRUD 및 공유기능을 수행한다.

1. 생성

- 1) 사용자는 상단 탭의 Main을 선택한다.
- 2) Main 페이지에서 Add project를 누른다.
- 3) 프로젝트를 함께 할 멤버를 선택한다.
- 4) 선택된 멤버에 한해서 초대 메일을 보낸다.
- 5) 프로젝트 생성을 완료한다.

2. 읽기, 갱신, 삭제

- 1) 사용자는 읽기, 갱신 또는 삭제할 프로젝트를 선택한다.
- 2) 프로그램은 해당 프로젝트에 대한 정보와 트리뷰를 보여준다.
- 3) 사용자는 트리뷰에서 읽기, 갱신 또는 삭제할 소스코드를 선택한다.
- 4) 사용자는 소스코드에 대해 읽기, 갱신 또는 삭제를 수행한다.
- 5) 프로그램은 해당 소스코드를 갱신한다.

3. 공유

- 1) 사용자는 공유할 프로젝트를 선택한다.
- 2) 프로그램은 공유를 위한 창을 보여준다.
- 3) 사용자는 공유받을 사용자의 e-mail 주소를 입력한다.
- 4) 프로그램은 해당 e-mail로 공유 메시지를 보낸다.

-이슈 관리

프로젝트에 대하여 이슈(프로젝트 상의 문제) 발생 시 CRUD 및 공유기능을 수행한다.

전제조건: 프로젝트에 관한 권한이 있어야한다.

1. 생성

- 1) 사용자는 상단 탭의 Main을 선택한다.
- 2) Main 페이지 좌측 탭의 Today's Issue를 선택한다.
- 3) 프로그램은 Issue 게시판을 보여준다.
- 4) 사용자는 해당 페이지에서 이슈를 생성한다.

2. 읽기, 갱신, 삭제

- 1) 사용자는 Issue 게시판에서 읽기, 갱신 또는 삭제할 이슈를 선택한다.
- 2) 사용자는 선택한 이슈에 대해 읽기, 갱신 또는 삭제를 수행한다.
- 3) 프로그램은 해당 Issue를 갱신한다.

3. 공유

- 1) 사용자는 공유할 Issue와 멤버를 선택한다.

2) 프로그램은 해당 Issue를 선택된 멤버에게 공유한다.

-룰 관리

코드 컨벤션에서 사용할 룰 라이브러리에 대해 확인 및 설정이 가능하다.

전제조건: 룰 셋팅을 위한 프로젝트가 존재해야 한다.

- 1) 사용자는 상단 탭의 Main을 선택한다.
- 2) Main 좌측 탭의 Rule Setting을 선택한다.
- 3) 프로그램은 Code-Convention 탭으로 넘어가고 룰 셋팅을 위한 창을 보여준다.
- 4) 사용자는 variable, function, statement 단위의 원하는 룰을 선택한다.
- 5) 프로그램은 해당 룰을 갱신한다.

-코드 컨벤션

주어진 프로젝트 및 소스에 대하여 지정된 룰을 이용하여 코드를 검증한다.

전제조건: 코드 컨벤션을 위한 룰 셋팅이 완료되어 있어야 한다.

- 1) 사용자는 상단 탭의 Code-Convention을 선택한다.
- 2) 사용자는 코드 컨벤션을 수행할 소스코드/ 프로젝트를 선택하고 실행한다.
- 3) 프로그램은 해당 소스코드/ 프로젝트에 관한 코드 컨벤션을 수행한다.
- 4) 사용자는 소스코드에 대해 읽기, 갱신 또는 삭제를 수행한다.
- 5) 프로그램은 결과에 대한 화면을 보여준다.

-코드 리뷰

주어진 소스를 확인하고 코멘트를 남긴다.

전제조건: 코드 리뷰를 수행할 프로젝트에 관한 권한이 있어야 한다.

- 1) 사용자는 상단 탭의 Code-Review를 선택한다.
- 2) 프로그램은 해당 프로젝트에 대한 정보와 트리뷰를 보여준다.
- 3) 사용자는 코드 리뷰할 소스코드를 서택한다.
- 4) 사용자는 코드를 확인하고 코멘트를 남기고, confrimed/ failed를 선택한다.
- 5) 프로그램은 해당 코드를 리뷰한 비율을 시각적으로 보여준다.

4. 작품 개발 환경

구분		상세내용
S/W 개발환경	OS	Window 10, Ubuntu 16.04 LTS
	개발환경(IDE)	VS 2017, VS Code,
	개발도구	Bootstrap, PostgreSQL, SonarQube, Ace, Codiad
	개발언어	HTML5&CSS3, JS, PHP
	기타사항	Autoset 9
H/W 구성장비	디바이스	
	센서	
	통신	
	개발언어	
	기타사항	
프로젝트 관리환경	형상관리	Github
	이슈관리	Github 및 카카오톡
	의사소통관리	카카오톡 및 디스코드
	기타사항	

III. 프로젝트 수행 내용

※ 평가항목 : 수행능력 (문제해결능력, 수행충실성)

1. 멘티(참여학생) 업무분장

번호	이름	대학	학과	학년	역할	담당업무
1	문성욱	부산대대학	정보컴퓨터공학	4학년	팀장	SW 설계 및 모듈 개발
2	박문수	부산대대학	정보컴퓨터공학	4학년	팀원	DB 모델링 및 UI 개발
3	권성욱	부산대대학	정보컴퓨터공학	4학년	팀원	UI 설계 및 SW 모듈 개발
4	박슬기	홍익대대학	컴퓨터공학	3학년	팀원	UI 디자인 및 개발
5	전효진	홍익대대학	컴퓨터공학	3학년	팀원	DB 모델링 및 개발 문서 작성

2. 프로젝트 수행일정

프로젝트 기간 (한이음 사이트 기준)		2017.00.00. ~ 2017.00.00.											
구분	추진내용	프로젝트 기간											
		1월	2월	3월	4월	5월	6월	7월	8월	9월	10월	11월	12월
계획	수행계획서 작성 및 프로젝트 관련 조사												
분석	요구 사항 정의서 정의												
	요구 사항 추적표 작성												
설계	WEB-UI 디자인 및 Mockup작성												
	WEB-모듈관련 조사 및 설계												
	DB 분석 및 설계												
개발	UI 개발(based on UI_Mockup)												
	WEB-모듈 컴포넌트식 개발												
	WEB-모듈 통합 개발												
	WEB-DB 모델링												
테스트	WEB-기능 테스트												
종료	개발 종료												

3. 프로젝트 추진 과정에서의 문제점 및 해결방안

3-1. 프로젝트 관리 측면

- 프로젝트 생성 : 프로젝트를 설계하고 관리를 하는 일이 대부분 팀원이 처음인 경우가 많았었고 개발 방향에 대한 문제가 생기게 되었음. 따라서 멘토님에게 이 부분에 대하여 상의 드리고 팀원간에 회의를 진행하며 현 개발 상황의 문제인식과 앞으로의 방향에 대하여 생각을 할 수 있었음. 더불어서 프로젝트 개발/언어에 대한 문서를 멘토님으로부터 팀이 받을수 있었던 점도 이부분을 해결하는 데 큰 도움이 되었음.
- 프로젝트 컴포넌트 개발 : 프로젝트를 컴포넌트식으로 분산하여 개발을 진행하는 경우가 이루어짐. 추후에 컴포넌트를 하나의 완성된 기능으로 병합을 하는 데 있어서 각자의 스타일이나 라이브러리 문제가 발생하였음. 이를 해결하기 위해 공통된 스타일 라이브러리(bootstrap, ace-builder)를 사용하기로 하였음.
- 팀원 간 작업 능률성 : 각자 선호하는 분야와 언어가 있는 상황에서 일을 분배하는 과정에서 중첩되거나 혹은 전혀 새로운 분야를 맡게 되는 경우가 종종 발생하였음. 때문에 작업의 능률이 떨어지는 상황이 생기게 되었음. 이를 해결하기 위해 팀장이 개인 역량을 아는 상황에서 2-3명씩 팀 내부에서도 조를 나눠서 개발을 진행하게되었음.

3-2. 작품 개발 측면

- 개발 도구 선택 : 개발 단계에 있어서 사용할 언어를 제약을 해놓은 상황이었지만 앞서 야기된 컴포넌트 통합 문제를 완벽히 해결을 하지 못할 상황이었기 때문에 프로젝트를 원활하게 진행하고 완성도를 높이기 위한 방법으로 개발 도구(Angular)를 새롭게 선택하게됨. 처음 진입장벽이 있었기 때문에 월 2-3회 온라인 스터디 미팅을 통하여 개발 도구에 대한 이해와 활용도를 높여나가게 되었음.

4. 프로젝트를 통해 배우거나 느낀 점

- 한이음 프로젝트에서 새로운 프로젝트에 대한 영감을 받을 수 있었습니다. 할당 받은 한이음 프로젝트의 주제는 학교에서 개념적으로만 배웠었고 쉽사리 와닿지 않는 내용이었습니다. 때문에 프로젝트에 대한 많은 회의 시간이 필요하였고 접목할 아이디어를 발의하고 그것을 발전시키는 경험을 반복적으로 할 수 있었습니다. 초기에 프로젝트 생성에 대한 경험을 깊이 새긴 뒤 다음으로 가장 많이 하려고 했고 실제로 많이 했던 것은 멘토:멘티, 멘타:멘티 간의 대화였습니다. 주기적으로 한달마다 멘토:멘티 모임 외에도 프로젝트 진행 및 개발을 위해 온라인 미팅과 스터디를 멘티간에 진행을 했습니다. 오프라인 미팅시에는 프로젝트 설계와 발전 방향을 멘토님으로부터 강의를 현장에서 들을 수 있었고 그 뒤 필요한 것에 대한 구상을 진행하고 팀 피드백을 갖는 시간을 가졌습니다. 온라인 미팅에서는 누가 무엇을 얼마나 성과물을 내었는 지 보다는 개발 단계마다 필요한 기능 정의와 정의된 이유, 목적을 하나로 통합하려는 시도를 가장 많이 했던 것 같습니다. 개발하는 데 있어서 뜻을 하나로 모으지 못하면 나중에 문제를 야기할 수 있다는 팀 판단 하에 위와 같이 진행했었습니다. 이와 같은 팀활동을 통해 문제 해결에 대한 통찰력 기를 수 있었고 방법론에 대해서 깊이 배울 수 있는 좋은 경험이었습니다. 실제로 한이음 프로젝트를 통해 배운 경험으로, 다른 프로젝트를 진행하는 데 있어서 문제 해결에 대한 진입속도를 향상시킬 수 있었다는 멘티의 이야기도 들을 수 있었습니다.
- 매달 한이음 프로젝트를 진행하면서 멘토 강의, 멘티간 스터디를 통해 남들보다 실무적으로 조금 더 앞서나가는 진취적인 느낌은 개발자로서의 수준 향상과 팀 사기에도 매우 고무적이었습니다. 그리고 한이음 데이와 같은 행사를 통해서 현재의 개발자와 미래의 개발자들간에 교류를 제공하는 자리 또한 프로젝트를 수행하는 데 있어서 좋은 자극이 되었습니다. 하나의 프로젝트를 완성함으로써 앞으로의 만들어갈 프로젝트에 대한 근거있는 자심감을 가질 수 있게 해준 소중한 기회이자 멋진 경험이었습니다.

IV. 작품의 기대효과 및 활용분야


※ 평가항목 : 기획력 (활용가능성)

1. 작품의 기대효과

- 클라우드 기반으로 동작하여 설치과정을 생략.
- 코드 컨벤션 및 코드 리뷰 기능 내장.
- 사용자 프로젝트가 웹에 저장되어 언제든지 사용가능.
- IDE의 기능을 교육용으로 집중 및 튜터와 튜티의 입장을 고려.

플랫폼/디바이스 독립성 - 어디서나 개발 가능

- 인터넷에 연결 가능한 데스크톱, 노트북, 태블릿만 있으면
- OS나 웹브라우저에 상관없이

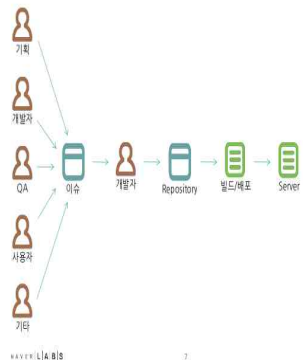


Google Java Style Guide

Table of Contents

- 1 Introduction
 - 1.1 Terminology, notes
 - 1.2 Guide notes
- 2 Source file basics
 - 2.1 File name
 - 2.2 File encoding, UTF-8
 - 2.3 Special characters
- 3 Source file structure
 - 3.1 License or copyright information, if present
 - 3.2 Package statement
 - 3.3 Import statements
 - 3.4 Class declaration
- 4 Formatting
 - 4.1 Braces
 - 4.2 Block indentation: +2 spaces
 - 4.3 One statement per line
- 4.6 Whitespace
- 4.7 Grouping parentheses: recommended
- 4.8 Specific constructs
- 5 Naming
 - 5.1 Rules common to all identifiers
 - 5.2 Rules by identifier type
 - 5.3 Camel case: defined
- 6 Programming Practices
 - 6.1 @Override: always used
 - 6.2 Caught exceptions: not ignored
 - 6.3 Static members: qualified using class
 - 6.4 Finalizers: not used
- 7 Javadoc
 - 7.1 Formatting
 - 7.2 The summary fragment
 - 7.3 Where Javadoc is used

어떻게 협업 하는가?




2. 작품의 활용분야

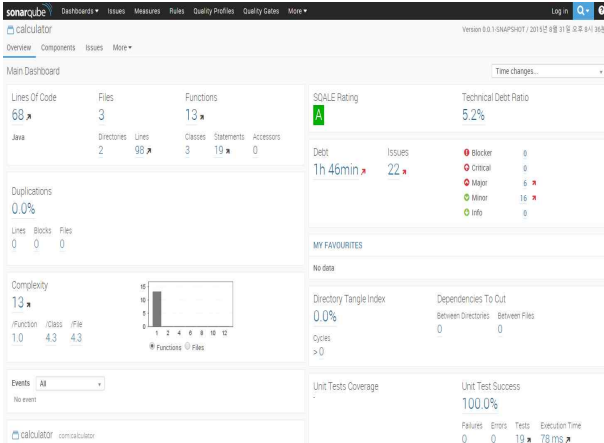
- 이용자가 표준 코드 스타일 가이드를 지키도록 유도.
- 사용자의 코드 스타일을 분석 후 통계 자료를 지속적으로 보고.
- 프로그램 설치 없이 클라우드 기반으로 동작하여 누구나 간편하게 사용가능.
- 코드 리뷰 기능을 이용한 손쉽게 협업을 할 수 있는 환경제공.

data types prefix example

data types	prefix	example
local	none	i8TempVar
global	g	gi8TempVar
signed	none	i8TempVar
unsigned	u	ui8TempVar
array	a	ai8TempVar
char_t	c	cTempVar
int8_t	i8	i8TempVar
int16_t	i16	i16TempVar
int24_t	i24	i24TempVar
int32_t	i32	i32TempVar
float_t	f	fTempVar
double_t	d	dTempVar
bit_t	b	bTempVar
bool_t	bl	blTempVar

코드리뷰 미팅

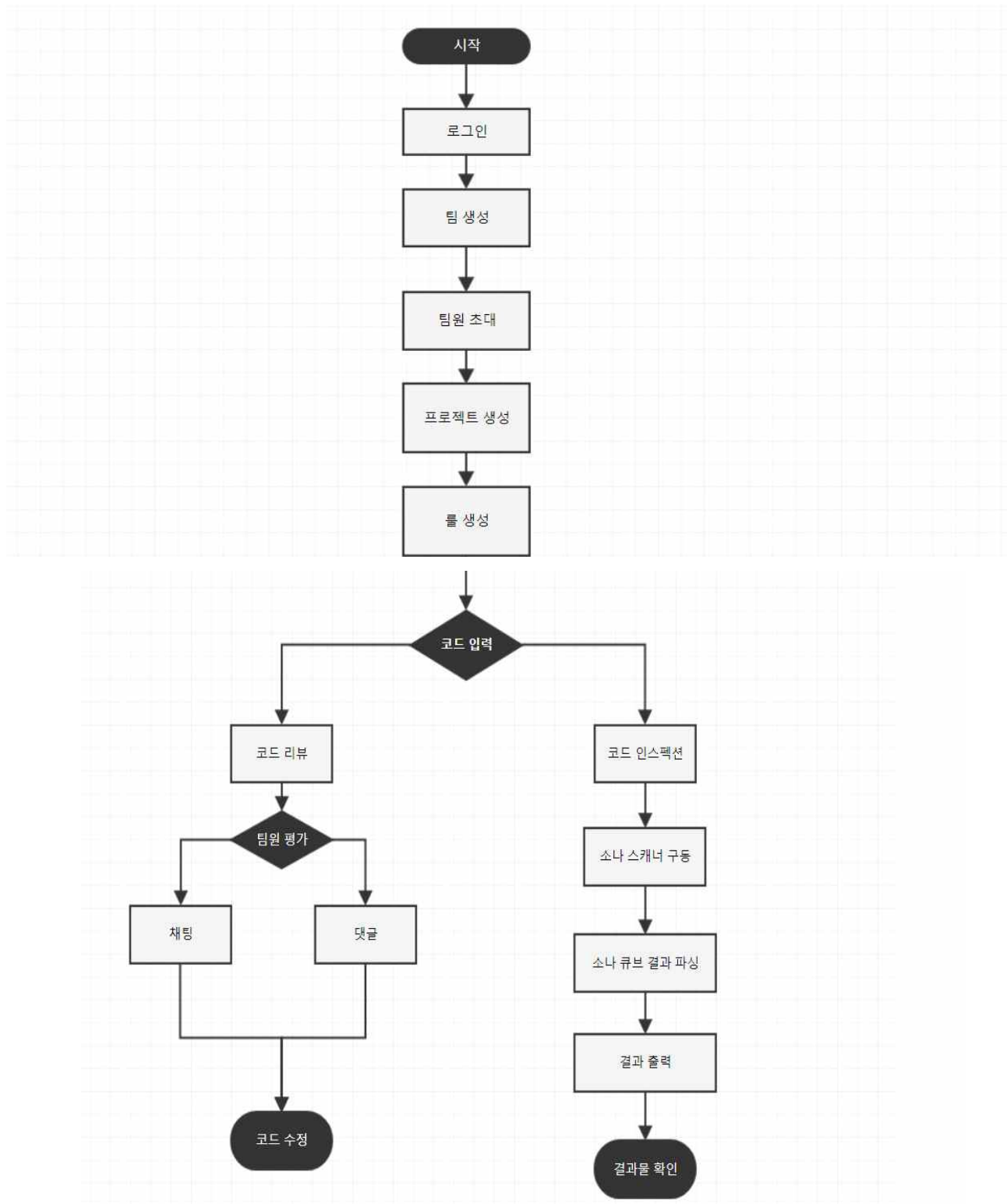




V. 개발산출물

※ 평가항목 : 평가 전반에 참고

○ 중요 Part Flow Chart 흐름도




```

define ("ace/lib/regexp",["require","exports","module"], function (require , exports
, module ) {
"use strict";

var real = {
exec:RegExp .prototype .exec ,
test:RegExp .prototype .test ,
match:String .prototype .match ,
replace:String .prototype .replace ,
split:String .prototype .split
},
compliantExecNpcg=real .exec .call (/(())?$/, "")[1 ] ===undefined , // check `exec`
handling of nonparticipating capturing groups
compliantLastIndexIncrement=function () {
var x = /^/g ;
real .test .call (x , "");
return !x .lastIndex ;
}();

if (compliantLastIndexIncrement&&compliantExecNpcg)
return ;
RegExp .prototype .exec =function (str ) {
var match =real .exec .apply (this , arguments ),
name , r2 ;
if ( typeof (str ) =='string'&&match ) {
if (!compliantExecNpcg&&match .length >1 &&indexOf (match , "") >-1 ) {
r2 =RegExp (this .source , real .replace .call (getNativeFlags (this ), "g", ""));
real .replace .call (str .slice (match .index ), r2 , function () {
for (var i =1 ; i <arguments .length -2 ; i ++) {
if (arguments [i ] ===undefined )
match [i ] =undefined ;
}
});
}
if (this ._xregexp &&this ._xregexp .captureNames ) {
for (var i =1 ; i <match .length ; i ++) {
name =this ._xregexp .captureNames [i -1 ];
if (name )
match [name ] =match [i ];
}
}
if (!compliantLastIndexIncrement&&this .global &&!match [0 ].length && (this
.lastIndex >match .index ))
this .lastIndex --;

```

```

    }
    return match ;
    };
    if (!compliantLastIndexIncrement) {
    RegExp.prototype.test =function (str ) {
    var match =real .exec .call (this , str );
    if (match &&this .global &&!match [0 ].length && (this .lastIndex >match .index ))
    this .lastIndex --;
    return !!match ;
    };
    }

    function getNativeFlags (regex ) {
    return (regex .global ?"g":"" ) +
    (regex .ignoreCase ?"i":"" ) +
    (regex .multiline ?"m":"" ) +
    (regex .extended ?"x":"" ) +// Proposed for ES4; included in AS3
    (regex .sticky ?"y":"" );
    }

    function indexOf (array , item , from ) {
    if (Array .prototype .indexOf ) // Use the native array method if available
    return array .indexOf (item , from );
    for (var i =from ||0 ; i <array .length ; i ++ ) {
    if (array [i ] ===item )
    return i ;
    }
    return -1 ;
    }

    });

```